Static Object Tracking in Road Panoramic Videos

Zhong Zhou[1,2], Ben Niu[1,2], Chen Ke[1,2], Wei Wu[1,2]
[1]State Key Laboratory of Virtual Reality Technology and System
[2]School of Computer Science and Engineering
Beihang University, Beijing, China
zz@vrlab.buaa.edu.cn

***Abstract*—In panoramic videos, the object movement between adjacent side images leads to deformation and discontinuity, which makes the traditional video tracking approaches insufficient. An effective static object tracking algorithm is proposed in this paper to resolve the tracking problems from the deformation and discontinuity in cubic panorama. The algorithm extends the relevant side images with boundary consistency, and then conducts a background eliminated mean-shift algorithm to track objects on the extended images. Experiment results show that the algorithm can track static objects correctly in reasonable situations in real-time.**

Keywords-panoramic video, object tracking, panorama expansion, Mean-Shift

I. INTRODUCTION

Object tracking in video is a challenging task with many applications such as surveillance, automatic video-indexing and traffic monitoring. According to the camera motion, the object tracking is classified into stationary camera-based tracking and ground-vehicle based tracking. The most common method from the stationary camera is to make a statistical model for the background. Stauffer et al. [1] is the first to use a mixture of Gaussians model the background, which is able to adapt to background changes such as swaying trees and flickering lights. Regions extracted as foreground is tracked between frames using Kalman filters. Object tracking from moving camera is more difficult because of the complex outdoor scenes which are always combined with rapidly changing illumination and blur effects.

It is not possible for a single camera to observe a large area in detail because of its finite field-of-view. Then the interests of using multiple cameras to track arise to get the depth information or extend the scope of view area. An important issue in using multiple cameras is the relationship between the different camera views which can be manually defined [2] or computed automatically [3] from the observations of the objects moving in the scene.

This paper focuses on the algorithm for tracking objects in panoramic video. The panoramic video covers 360°×180° view of scenes, and image deformation makes it difficult to track objects in sphere-based panoramic videos. However, the cubic panorama is suitable for tracking as it consists of six side images that are regular planar as shown in Fig. 1. Unfortunately little work has been done of tracking in such panoramic videos with dynamic background as road panorama.

This paper is organized as follows: the second section describes related work of object tracking in large area; the third part introduces the algorithm of tracking with the improved mean-shift on expanded cubic panorama; the next section describes the experiments and results analysis; and the last part is conclusion.

II. RELATED WORK

Several algorithms for object tracking from multiple stationary cameras have been proposed. Common methods emerged such as constructing blobs in 3D space using short-base line stereo matching with multiple stereo cameras [4], or using volume intersection [5]. Lee et al. [6] align the ground plane across multiple views to build common coordinates for multiple cameras. An automated surveillance system proposed by Lim et al. uses multiple PTZ (pan-tilt-zoom) cameras to track object in a wide scene [7]. But these approaches are suitable for moving object surveillance with stationary cameras. Patil et al. [8] made use of a combination of frame differencing, face detection and adaptive color blob tracking based on mean shift analysis to detect and track people in the panoramic image. But the algorithm is aiming at usage in meeting environments, and requires static background.

The aforementioned multi-camera tracking methods assume stationary cameras. Kang et al. [9] use a combination of stationary and pan-tilt-zoom cameras with overlapping views for tracking. However, it is not possible to have overlapping camera views due to orthogonality of the adjacent side images in cubic panorama. Methods for tracking without overlapping views in such a scenario inherently have to deal with sparse object observations.

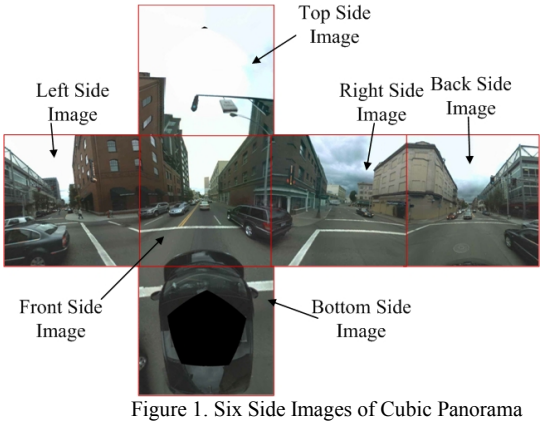

Figure 1. Six Side Images of Cubic Panorama

978-0-7695-4217-1/10 $26.00 © 2010 IEEE
DOI 10.1109/ISM.2010.18

Therefore some assumptions are made about the object speed and the path in order to obtain the correspondences across cameras [10]. The performance of these algorithms depends greatly on how much the objects follow the established paths and expected time intervals across cameras. For scenarios the spatio-temporal constraints cannot be used.

However, current tracking approaches do not have solution to tracking the object in cubic panoramic video when it moves between side images. When the object moves between adjacent side images, the deformation will occur (Fig. 2(a)). It is even worse when the movement affects three side images that some discontinuities may appear (Fig. 2(b)). The deformation and discontinuities lead to inefficiency of tracking in common situations or failure at the worst.

This paper presents an effective object tracking algorithm for cubic panoramic videos in order to solve these deformation and discontinuity problems. According to the features of the object motion in panoramic videos, the side images are expanded with boundary consistency. Upon the expanded image, an improved mean-shift algorithm is proposed to track the object.

III. ALGORITHM

A. Main Idea

The panoramic video covers 360°×180° view of scenes where the vehicle-mounted camera follows the road, moving in the direction from the back side to the front. Considering the object movement between adjacent sides, this particular motion can be utilized for the tracking. The epipolar lines of the panoramic video of static scenes [11] can be obtained as shown in Fig. 3, which describe the motions of static scene pixels. They radiate at the epipole on the front side image, move approximately in the horizontal direction in the top, bottom, left and right side images and finally converge at the epipole on the back side image. The movement of the epipolar lines reflects the motion of the panoramic camera. Between adjacent frames, the motion of the camera is mostly horizontal.

The movement of an object beside the road is exemplified in Fig. 4, which follows the directions of the epipolar lines. Assume tracking target is the line segment *AB*. Unfortunately, when the camera goes further, part of the target *AB* gets a transition and consequently is bent to be perpendicular to the boundary between the front and the top image. This deformation affects the object subsequent tracking. To reduce this distortion, *AB* will be curved appro-

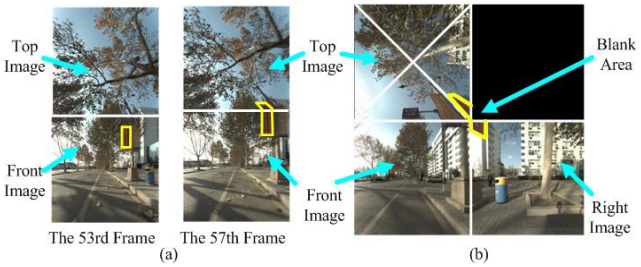

Figure 2. The deformation and discontinuity: (a) The deformation between the two adjacent images (b) The discontinuity at the corner of three side images

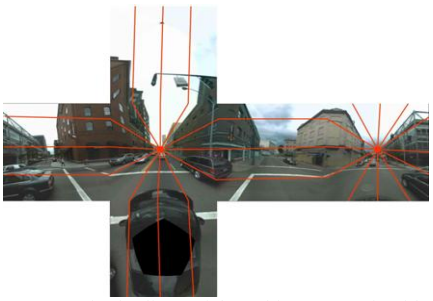

Figure 3. Epipolar lines of the cubic panoramic video

ximately along *OC* as much as possible. So an algorithm is proposed in this paper which conducts background eliminated mean-shift method on an expansion of cubic panorama. The proposed algorithm is summarized as follows:

Step 1) In the initial frame, manually select the region of the target or automatically detect the object to be tracked. The motion vector of the object is initialized as 0.

Step 2) Acquire a new frame. According to the epipolar lines and the motion vector of the target, expand the reference side image with the adjacent images on the cube.

Step 3) The background eliminated mean-shift algorithm starts on the expanded images, and the motion vector of the object is updated.

Step 4) If the object still moves on the boundaries between side images, go to Step 2); otherwise, utilize the 在 improved mean-shift algorithm on the ordinary side images of the subsequent frame.

B. Side Expansion for Cubic Panorama

Unlike the regular videos, cubic panorama has several side images. Therefore the tracking needs to know where an object goes to from one image. An expansion is designed for cubic panorama to benefit the image continuity for tracking as Fig. 5. Both of the top and the bottom images are split into four triangle parts as *S1*-*S4* and *S5*-*S8* in Fig. 5 respectively. We would like the expansion to preserve the consistency of object shape after the padding. We define $''\Rightarrow''$ as the expansion operator which means the left region of the operator is expanded to fill the right one. Side expansion for cubic panorama is defined as:

$$\begin{aligned} &S1\Rightarrow NPQF,\ S2\Rightarrow PBSQ,\ S3\Rightarrow ANFE,\ S4\Rightarrow BRTS \\ &S5\Rightarrow B_1R_1T_1S_1,\ S6\Rightarrow A_1N_1F_1E_1,\ S7\Rightarrow P_1B_1Q_1S_1,\ S8\Rightarrow N_1P_1Q_1F_1 \end{aligned} \quad (1)$$

An example is illustrated of triangle parts *S1* and *S2* of the top image. The *S1* part of the top image is stitched to the

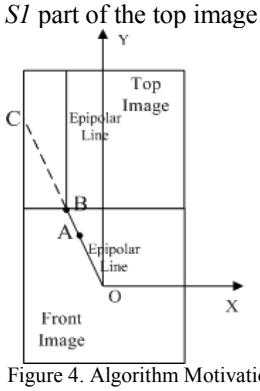

Figure 4. Algorithm Motivation

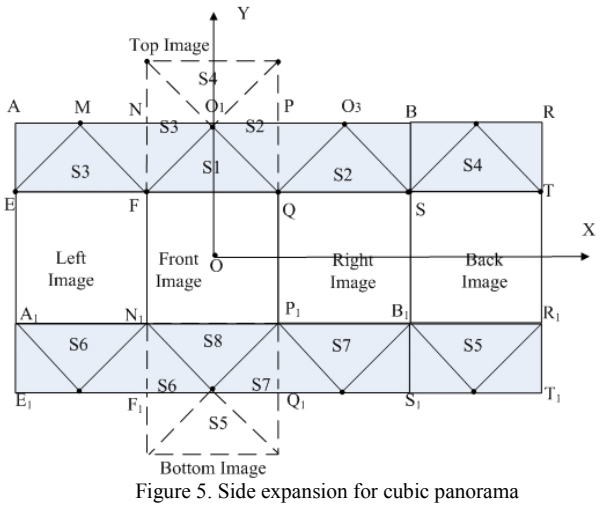

Figure 5. Side expansion for cubic panorama

edge *FQ* and is expanded to fill the area *NPQF*. Similarly, the *S2* part of the top image is stitched to the edge *QS* and is expanded to fill the area *PBSQ*. Geometrically, the edge QO_1 and the edge QO_3 are contiguous. Consequently, the expansion of the area *NPQF* and area *PBSQ* preserves the pixel continuity on their edges.

To present the procedure of the expansion, we have six side images pixel sets *Top*, *Left*, *Front*, *Right*, *Back*, and *Bottom* as shown in Fig. 5.

$$\begin{aligned}
Top &= \left\{(x,y)\,|\left(-L/2 \le x \le L/2\right) \wedge \left(L/2 < y \le 3L/2\right)\right\} \\
Left &= \left\{(x,y)\,|\left(-3L/2 \le x \le -L/2\right) \wedge \left(-L/2 \le y < L/2\right)\right\} \\
Front &= \left\{(x,y)\,|\left(-L/2 \le x \le L/2\right) \wedge \left(-L/2 \le y < L/2\right)\right\} \\
Right &= \left\{(x,y)\,|\left(L/2 \le x \le 3L/2\right) \wedge \left(-L/2 \le y < L/2\right)\right\} \\
Back &= \left\{(x,y)\,|\left(3L/2 \le x \le 5L/2\right) \wedge \left(-L/2 \le y < L/2\right)\right\} \\
Bottom &= \left\{(x,y)\,|\left(-L/2 \le x \le L/2\right) \wedge \left(-3L/2 \le y < -L/2\right)\right\}
\end{aligned} \tag{2}$$

In (2) *L* indicates the edge size of the cubic panorama. Let *Surrounding* be the union of *Left*, *Right*, *Front* and *Back*:

$$Surrounding = Left \cup Front \cup Right \cup Back \tag{3}$$

In order to illustrate the transition of the expansion, we define a function $F : Cubic \to Rect$, where a pixel $C(x_c,y_c)$ is in the set *Cubic* and a pixel $K(x_k,y_k)$ is in the set *Rect*.

$$Cubic = Top \cup Surrounding \cup Bottom \tag{4}$$

$$Rect = \left\{(x_k,y_k)\,|\left(-3L/2 \le x_k \le 5L/2\right) \wedge \left(-L \le y_k \le L\right)\right\} \tag{5}$$

The following shows the derivation of the function F:

1) Obviously, the function F for $(x_c,y_c) \in Surrounding$ is given by:

$$F(x_c,y_c) = [x_k,y_k] = [x_c,y_c] \quad (x_c,y_c) \in Surrounding \tag{6}$$

2) For $(x_c,y_c) \in Top$, our method utilizes a triangle area to fill a rectangle area by padding the rectangle with pixels from the triangle. Each vertical line of the rectangle is padded with pixels in an oblique edge of the triangle. This pixel padding is exemplified by the condition $(x_c,y_c) \in S1$ and $(x_c,y_c) \in S2$, as shown in Fig. 6.

a) In the case of $(x_c,y_c) \in S1$, a pixel $K(x_k, y_k)$ of the rectangle area *NPQF* is filled with a pixel $C(x_c,y_c)$ of triangle part *S1* of the top image, as shown in Fig. 6(a). The perpendicular projection of $K(x_k, y_k)$ to X axis intersects the boundary *FQ* at $H(x_k, L/2)$. The pixel $C(x_c,y_c)$ is on the line HO_1 and $y_c = y_k$, so the equation of a straight line HO_1 is calculated by

$$\frac{y - L}{x} = \frac{y_c - L}{x_c} \tag{7}$$

From (7), the coordinate of the pixel $K(x_k, y_k)$ is given by

$$\begin{cases} x_k = \dfrac{L}{2} \cdot \left(\dfrac{x_c}{L - y_c}\right), & y_c \ne L \\ y_k = y_c \end{cases} \tag{8}$$

In (8) *L* is also the edge size of the side image. Hence we can write

$$F(x_c,y_c) = [x_k,y_k] = \begin{cases} \left[\dfrac{L}{2} \cdot \left(\dfrac{x_c}{L - y_c}\right), y_c\right] & y_c \ne L \\ [0,L] & y_c = L \end{cases}, \quad (x_c,y_c) \in S1 \tag{9}$$

b) For $(x_c,y_c) \in S2$, $K'(x_{k'}, y_{k'})$ is on the line HO_3 and $y_{k'} = y_k$, as shown in Fig. 6(b). In accordance with the expansion of S1, $K(x_k, y_k)$ is calculated by

$$\begin{cases} x_k = \dfrac{L}{2} \cdot \left(\dfrac{x_{k'} - 2y_{k'} + L}{L - y_{k'}}\right) \\ y_k = y_{k'} \end{cases} \tag{10}$$

Let *G* be the transformation group of translation and rotations, such that the point *C* can be transformed from the point *K'* by (11) in Fig. 5.

$$\begin{aligned}
[x_c,y_c,1] &= [x_{k'},y_{k'},1] \cdot G \\
&= [x_{k'},y_{k'},1] \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -L & 1 \end{bmatrix} \begin{bmatrix} \cos(-90^\circ) & \sin(-90^\circ) & 0 \\ -\sin(-90^\circ) & \cos(-90^\circ) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ L & L & 1 \end{bmatrix}
\end{aligned} \tag{11}$$

By imposing (10), we obtain

$$F(x_c,y_c) = \left[\frac{L}{2} \cdot \left(\frac{y_c - 2x_c + 2L}{L - x_c}\right), x_c\right] \quad (x_c,y_c) \in S2 \tag{12}$$

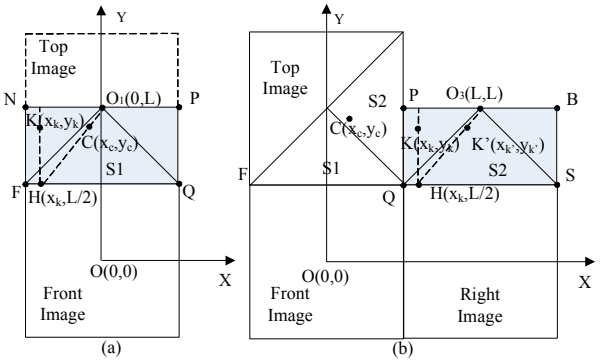

Figure 6. The pixel calculation in the expansion: (a) expansion of the front image (b) expansion of the right image

c) In the same way, the function F is respectively calculated by Equation (13) and (14) on the condition that $(x_c,y_c)\in S3$ and $(x_c,y_c)\in S4$:

$$F(x_c,y_c)=\left[\frac{L}{2}\cdot\left(\frac{y_c-2x_c-2L}{L+x_c}\right),-x_c\right] \quad (x_c,y_c)\in S3 \tag{13}$$

$$F(x_c,y_c)=\begin{cases}\left[\frac{L}{2}\cdot\left(\frac{-x_c+4y_c-4L}{y_c-L}\right),2L-y_c\right] & y_c\neq L\\ [0,L] & y_c=L\end{cases},(x_c,y_c)\in S4 \tag{14}$$

d) In similarity, we obtain the function F for $(x_c,y_c)\in S5,S6,S7,S8$, as follows:

$$F(x_c,y_c)=\begin{cases}\left[\frac{L}{2}\cdot\left(\frac{-x_c-4y_c-4L}{-y_c-L}\right),2L+y_c\right] & y_c\neq -L\\ [0,-L] & y_c=-L\end{cases},(x_c,y_c)\in S5 \tag{15}$$

$$F(x_c,y_c)=\left[\frac{L}{2}\cdot\left(\frac{-y_c-2x_c-2L}{L+x_c}\right),-x_c\right] \quad (x_c,y_c)\in S6 \tag{16}$$

$$F(x_c,y_c)=\left[\frac{L}{2}\cdot\left(\frac{-y_c-2x_c+2L}{L-x_c}\right),x_c\right] \quad (x_c,y_c)\in S7 \tag{17}$$

$$F(x_c,y_c)=\begin{cases}\left[\frac{L}{2}\cdot\left(\frac{x_c}{L+y_c}\right),-y_c\right] & y_c\neq -L\\ [0,-L] & y_c=-L\end{cases},(x_c,y_c)\in S8 \tag{18}$$

Fig. 7(a) and Fig. 7(b) show the front and the right side image after expanding respectively. The expansions for cubic panorama rectify the deformation when objects move between adjacent side images and recover the discontinuousness at the corner of the cube.

C. Background Eliminated Mean-Shift Tracking

Mean-shift is a semiautomatic tracking algorithm that needs an automated detection or a manual method to localize the objects in the initial frame. It applies a rectangle or circular template to label the objects in the tracking process. However, this rectangle or circular template would introduce the color of background region into the calculation of histograms when it comes to dynamic background without subtraction. It will affect the correctness of object tracking in road panorama. A background eliminated mean-shift tracking is presented in this part.

To represent the target model, the mean-shift tracking method [12] defines $\{x_i^*\}_{i=1...n}$ as the locations of the target model and a function $b:R^2\rightarrow\{1...m\}$. The function associates the pixel at location x_i^* and the corresponding index $b(x_i^*)$ of the histogram bin to the color of the pixel. The algorithm applies the Epanechnikov profile for the histogram computation. The probability of the color u in the target model is calculated by

$$\hat{q}_u=C\sum_{i=1}^{n}k(||x_i^*||^2)\delta[b(x_i^*)-u],\ u=1...m \tag{19}$$

In (19) δ is the Kronecker delta function and C is the normalization constant derived by imposing the condition $\sum_{u=1}^{m}\hat{q}_u=1$; from where

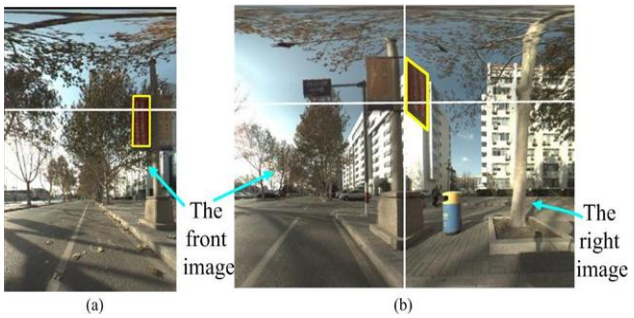

Figure 7. The expanded side images: (a) The expansion of the front side image (b) The expansion of the front image and the right image with a corner inside

$$C=\frac{1}{\sum_{i=1}^{n}k\left(\left|x_i^*\right|^2\right)} \tag{20}$$

Our approach involves learning a statistical color model of the background, which is used for segmenting the object that appears in foreground [13]. Each background pixel value is modeled as a multi-dimensional Gaussian distribution in HSV space, characterized by its mean value μ and standard deviation σ. Each color component $color_x$ is compared to the current distribution in order to mark foreground colors:

$$(color_x-\mu)^2>(2\sigma)^2 \tag{21}$$

The Gaussian distribution is updated for each color as follows:

$$\begin{aligned}\mu&\leftarrow\alpha\cdot color_x+(1-\alpha)\mu\\ \sigma^2&\leftarrow max\left(\sigma_{min}^2,\alpha(color_x-\mu)^2+(1-\alpha)\sigma^2\right)\end{aligned} \tag{22}$$

After the foreground colors have been marked, the colors of the target object, $color_1$, $color_2$, ... $color_m$, have been determined. Then m kinds of target colors are defined as from *1* to m. Consequently, the background colors from cb_1, cb_2, ..., cb_{M-m} are all set to be 0 to eliminate the impact of background color on the statistics histogram. The definition of target colors and background colors is formulated as:

$$\begin{cases}color_i=i & 1\le i\le m\\ cb_k=0 & 1\le k\le M-m\end{cases} \tag{23}$$

The elimination of background colors changes the M-dimension histogram into an (m+1)-dimensional histogram from 0 to m. The probability of background colors are set to 0 by

$$\hat{q}_{cb_k}=0\ ,\ k=1,2,...M-m \tag{24}$$

So the following equation is used for new quantization:

$$b^*(x_i)=\begin{cases}k & b(x_i)=color_k\\ 0 & else\end{cases} \tag{25}$$

The zero probability of background colors is equal to the elimination of background colors from the search window. An irregular shaped search window can also be extracted along the target contour. This background color elimination, which preserves the probability condition $\sum_{u=0}^{m}\hat{q}_u=1$, would

reduce the influence of dynamic background in the object tracking.

D. Discussion

For the expansion of *S1*, as shown in Fig. 8, the epipolar line *OU* and *VU* are bent on the boundary of the top side and the front side images. The proposed expansion method rectifies this swerve of epipolar lines and keeps the boundary consistency. Generally speaking, the closer the expanded *K* is to the line *OW*, the smaller the object deformation. The expanded points of *UV* construct a curve which starts from *U*. The accuracy of the rectification depends on the difference between the gradient of *OU* and the gradient of *K* on the curve.

From Fig. 8, the curve *f(y)* is rectified by the epipolar line *UV*, and the equation of the curve is calculated by

$$f(y)=\frac{Lx_{k'}}{2}\cdot\frac{1}{(L-y)} \quad , \qquad y\neq L \tag{26}$$

And the derivative of (26) is

$$f'(y)=\frac{Lx_{k'}}{2}\cdot\frac{1}{(L-y)^2} \quad , \qquad y\neq L \tag{27}$$

In (26) and (27) $K'(x_{k'}, y_{k'})$ denotes a pixel of triangle the part *S1* of the top image. The gradients of the line *UO* is $k_{UO}=2x_{k'}/L$, so we obtain

$$\lim_{y\to L/2} f'(y)=2x_{k'}/L=k_{UO} \tag{28}$$

As shown in (22), the closer *K* is to *FQ*, the more accurate the rectification. In Fig. 8, $W(x_w, y_w)$ is on the epipolar line *UO* and $y_w = y_k$. We calculate the Euclidean Distance in pixels between the expanded *K* and the point *W* as the accuracy of the rectification. In our experiment, the resolution of one side image of the cubic panorama is 512×512 and the expanded area *NPQF* includes 512×256 pixels. The experiment computed the sums of points of different accuracies, as shown in Table I.

In Table I, the points with the accuracy less than or equal to 5 pixels covers the 22.3% of the expanded area *NPQF* and the accuracy distribution of this area is illustrated in Fig. 9. In the expanded area, different colors denote points of different accuracies and the points with the accuracy more than 5 pixels are represented by black. In the expanded images, the closer the pixel is to the boundary, the more accurate the rectification. When a target moves onto the boundaries of the cube, the proposed algorithm can reduce the object distortion and track the target effectively.

TABLE I. STATISTICS OF POINTS WITH DIFFERENT ACCURACY

Statistics	Accuracy (In Pixels)				
	1	*2*	*3*	*4*	*5*
Sums of Points	17392	4098	3100	2532	2152
Total Points	256×512=131072				
Ratio	13.3%	3.1%	2.4%	1.9%	1.6%

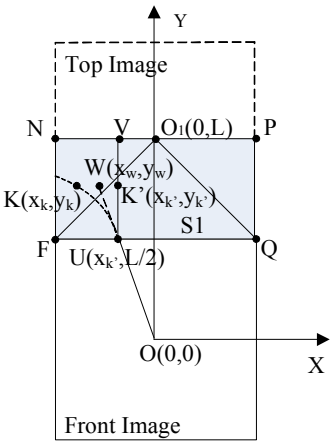

Figure 8. The rectification of the epipolar line

IV. EXPERIMENT ANALYSIS

The panoramic video used in our experiment is captured by Ladybug3 panoramic device along the campus road of Beihang University. The resolution of each side image in a frame is 512×512 and frame rate is 15 fps. The experiment expands the top side image of the cubic panorama for example.

The experiment host is with CPU Intel core2 duo 2.66GHz with memory 2G. The average tracking time is 31.26 milliseconds per frame. We can have real-time object tracking for panoramic videos with the algorithm.

We have 8 panoramic video sequences and use 10 clips in them as samples. 10 static objects of different shapes and colors, includes 5 traffic signs (TS), 2 cars, 2 billboards (BD) and 1 building. The details of the tracking object information are shown in Table II, where "Affecting Side Images" illustrates the set of side images on which the object moves. In the initial frame, the locations of all traffic signs in the panorama are automatically selected by the detection algorithm discussed in reference [14] and others are selected manually. We use the Hue component in HSV color space and 64 levels for quantification.

In Fig. 10, several tracking results are displayed to compare. Fig. 10(a) shows the tracking results on the original sequence. The left image illustrates that the rectangle template should be extended to label the target due to the deformation, but it also introduces non-target area. The right

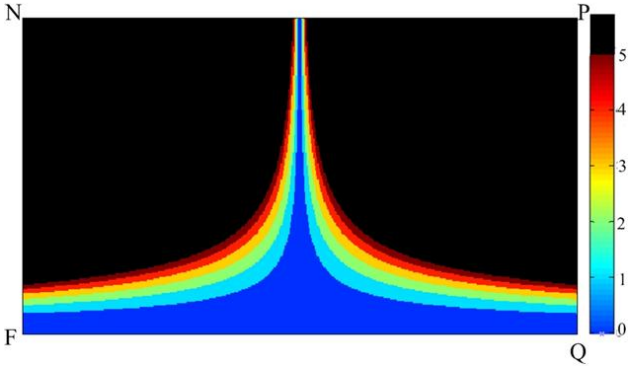

Figure 9. The rectification accuracy distribution of the area NPQF

TABLE II. OBJECT INFORMATION

Object	Accuracy (In Pixels)		
	Object Color	*Object Shape*	*Affecting Side Images*
TS1	Blue	Rectangle	Front, Right, Top
TS2	Yellow	Triangle	Front, Top
TS3	Blue	Rectangle	Front, Right
TS4	Red	Circle	Front, Right, Top
TS5	Blue	Circle	Front, Left, Top
Car1	Red	Irregular	Front, Right
Car2	Red	Irregular	Front, Right
BD1	Red	Rectangle	Front, Right, Top
BD2	Yellow	Rectangle	Front, Left, Top
Building	Grey	Irregular	Front, Right, Top

image shows the result of the tracking failure when the object moves at the corner of the cube. Fig. 10(b) presents the results on the expanded panoramic video. Compared to the Fig. 10(a), the proposed algorithm can effectively reduce the object distortion and successfully track the target.

To evaluate the performance of the algorithm, we choose Precision Ratio and Hit Ratio as the main experiment indices. The experiment counts the number of pixels that are both in the tracked region run by our algorithm and the hand-labeled ground truth of the target region. The ratio of tracked pixels to the ground-truth pixels is defined as the Precision Ratio, which reflects the accuracy of tracking results.

So Precision Ratio = j/i, where j is the number of actually tracked pixels and i is that of the target region. Considering the tracking is not for contour labeling but for object finding, 60% is chosen as the threshold whether the object is correctly tracked.

We have Hit Ratio as Hit Ratio=m/n, where m denotes the number of frames that are correctly tracked and n denotes the number of total frames. The panorama video clip is selected randomly, so the object maybe vanishes in the frames. Although being already vanished, we define the following frames non-hit.

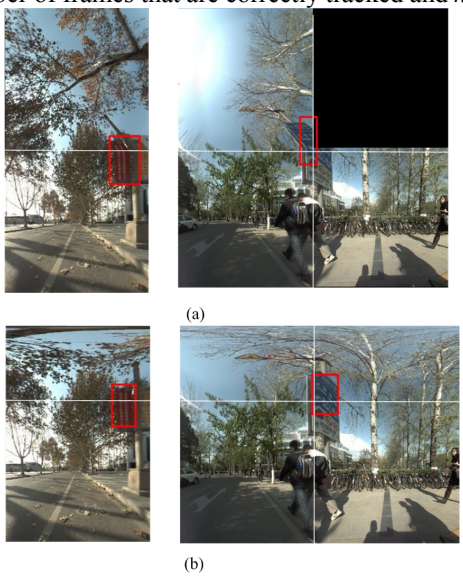

Figure 10. Comparison of the tracking results on the cubic panorama without expansion and with expansion (a) tracking on the 56th frame of the billboard sequence and the 74th frame of the traffic sign sequence (b) tracking on the expanded ones

The average Precision Ratio and Hit Ratio are shown in Fig. 11. For most samples, Precision Ratio are higher than 60%. The Hit Ratio seems to be high, but need further analysis because the non-hit frames may have vanishing target region or occlusions.

The samples have frames between 80 and 200. We have every five frames of the sequences manually for a quantitative assessment. As to these 1/5 frames, Fig. 12 presents the total frame number of the sample frames and the number of non-hit frames in it. The experiment totally samples 252 frames of 10 samples and the total number of non-hit frames is 82.

Furthermore, we analyze the non-hit reasons to find out how many is correct and how many is not, as shown in Fig. 13. The frame statistics is divided into 5 categories in the end. As to the total 82 non-hit frames, we have:

1) Class A means Vanishing: the size of the boundary box of the object is less than 5×5. Total 16 frames in Class A;

2) Class B means color mixture: the object goes into the background with similar color. Total 22 frames in Class B;

3) Class C means object missing because of the frame dropping. The frame drop occurs randomly during the Ladybug capture because of the resolution and laptop ability. Total 20 frames in Class C;

4) Class D means object missing for the occlusion. Total 10 frames in Class D;

5) Class E means the Hue component of color changes too quickly to more than 4 units. Totally 14 frames are in Class E.

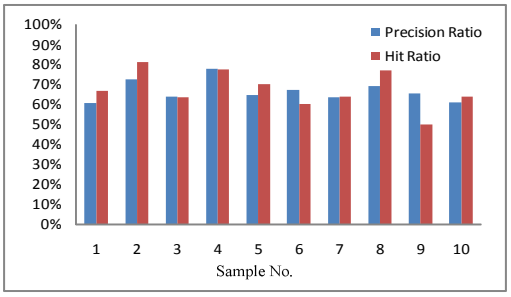

Figure 11. Average Precision Ratio and Hit Ratio

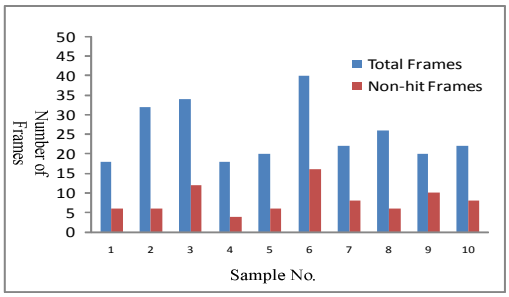

Figure 12. Total tested frames vs. Non-hit frames

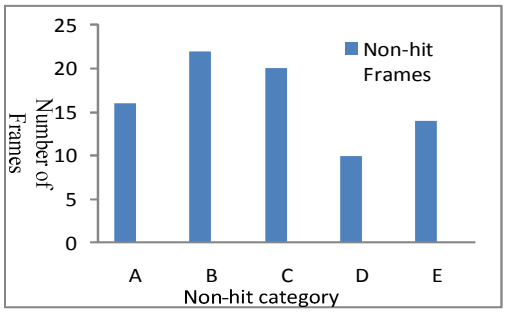

Figure 13. Non-hit category analysis

Fig. 13 shows the reason of non-hit frames classified by different categories. We think for the five classes, class A, B, D is reasonable for the non-hit. Class C may be improved to use high performance computer instead of the laptop for panorama recording. For class D, extra detection may help to find the object again to some extent if the object appearance does not change much. Class E frames may be the actual failure of object tracking.

Some tracking results are illustrated in appendix. Experiment results show that the algorithm can track the static objects correctly in cubic panoramic videos.

V. CONCLUSION

A static object tracking algorithm for cubic panoramic video is presented in this paper. The main contributions of the proposed algorithm are the cubic panorama expansion and the background eliminated mean-shift algorithm for object tracking. The side image is expanded according to the features of the object motion in panoramic video. Upon the expanded panorama, the background eliminated mean-shift algorithm is applied to track the static object based on the color information in the sequence. The experiment results illustrate that the proposed algorithm can track static objects correctly in reasonable situations in real-time.

ACKNOWLEDGMENT

This work is supported by the National Grand Fundamental Research 973 Program of China under Grant No. 2009CB320805, National Science & Technology Supporting Program No. 2008BAH37B08 and the Industry-Academy-Research Program of Guangdong Province & the Ministry of Education under Grant No. 2008A090400020, the Fundamental Research Funds for the Central Universities of China.

REFERENCES

[1] C. Stauffer, W. Eric, and L. Grimson, "Learning patterns of activity using real-time tracking," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 22, Aug. 2000, pp. 747-757, doi: 10.1109/34.868677.

[2] R.T. Collins, A.J. Lipton, H. Fujiyoushi, and T. Kanade, "Algorithms for Cooperative multisensor surveillance," Proc. IEEE, Aug. 2002, pp. 1456-1477, doi: 10.1109/5.959341.

[3] S. Khan, and M.Shah, "Consistent labeling of tracked objects in multiple cameras with overlapping fields of view," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 25, Sep. 2003, pp. 1355-1360, doi: 10.1109/TPAMI.2003.1233912.

[4] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale and S. Shafer, "Multi-camera Multi-person Tracking for Easy-Living," Proc. 3rd Workshop. Visual Surveillance (VS 2000), Aug. 2002, pp. 3-10, doi: 10.1109/VS.2000.856852.

[5] A. Mittal and L. S. Davis, "M2Tracker: A Multi-View Approach to Segmenting and Tracking People in a Cluttered Scene Using Region-Based Stereo," International Journal of Computer Vision, vol. 51, Feb. 2003, pp. 189-203, doi: 10.1023/A:1021849801764.

[6] L. Lee, R. Romano, and G. Stein, "Monitoring activities from multiple video streams: establishing a common coordinate frame," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 22, Aug. 2000, pp. 758-766, doi: 10.1109/34.868678.

[7] S. Lim, A. Elgammal, and L. S. Davis, "Image-based Pan-tilt Camera Control in a Multi-Camera Surveillance Environment," Proc. International Conf. on Multimedia and Expo(ICME 2003), IEEE Computer Society, Jul. 2003, pp. 645-648, doi: 10.1109/ICME.2003.1221000.

[8] R. Patil, P.E. Rybski, T. Kanade, and M. M. Veloso, "People detection and tracking in high resolution panoramic video mosaic," IEEE/RSJ International Conf. on Intelligent Robots and Systems(IROS 2004), Sep. 2004, pp. 1323-1328, doi: 10.1109/IROS.2004.1389579.

[9] J. Kang, I. Cohen, and G. Medioni, "Continuous tracking within and across camera streams," IEEE Conf. on Computer Vision and Pattern Recognition(CVPR 2003), Jun. 2003, pp. 267-272, doi: 10.1109/CVPR.2003.1211363.

[10] O. Javed, Z. Rasheed, K. Shafique, and M. Shah, "Tracking across multiple cameras with disjoint views," IEEE Conf. on Computer Vision(ICCV 2003), Oct. 2003, pp. 952-957, doi: 10.1109/ICCV.2003.1238451.

[11] F. Kangni, and R. Laganière, "Epipolar Geometry for the Rectification of Cubic Panoramas," Proc. 3rd Conf. on Computer and Robot Vision(CRV 2006), Jun. 2006, pp. 70-77, doi: 10.1109/CRV.2006.29.

[12] C. Dorin, R. Visvanathan, and M. Peter, "Real-time Tracking of Non-rigid Objects using Mean Shift," Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR 2000), Jun. 2000, pp. 142-149, doi: 10.1109/CVPR.2000.854761.

[13] M. Nicolescu, G. Medioni, and M. Lee, "Segmentation, Tracking and Interpretation Using Panoramic Video," Proc. IEEE Workshop on Omnidirectional Vision (OMNVIS 2002), Aug. 2002, pp. 169-174, doi: 10.1109/OMNVIS.2000.853826.

[14] W. Liu, X Chen and B. B. Duan, "A System for Road Sign Detection, Recognition and Tracking Based on Multi-cues Hybrid," IEEE Symp. Intelligent Vehicles (IVS 2009), Jun. 2009, pp. 562-567, doi : 10.1109/IVS.2009.5164339.

APPENDIX

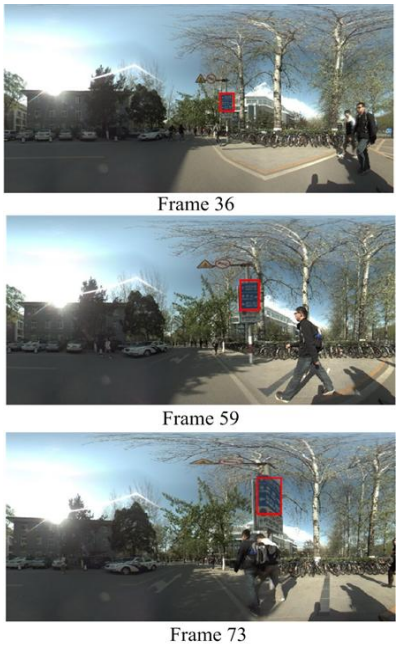

Figure 14. TS1 Sequnece

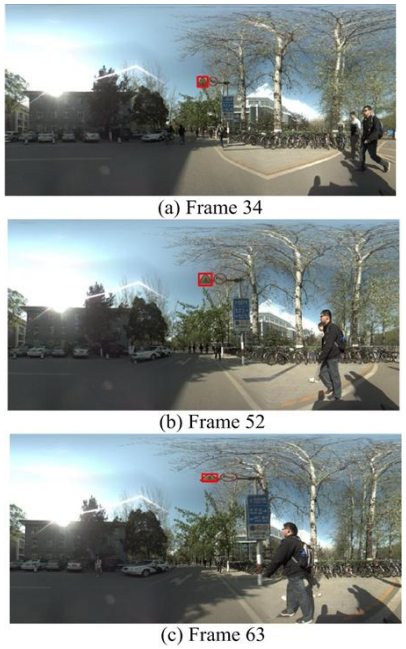

Figure 15. TS2 Sequence

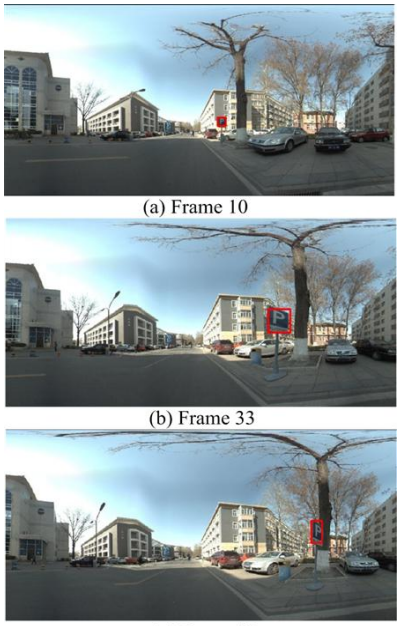

Figure 16. TS3 Sequence

Figure 17. Car1 Sequence

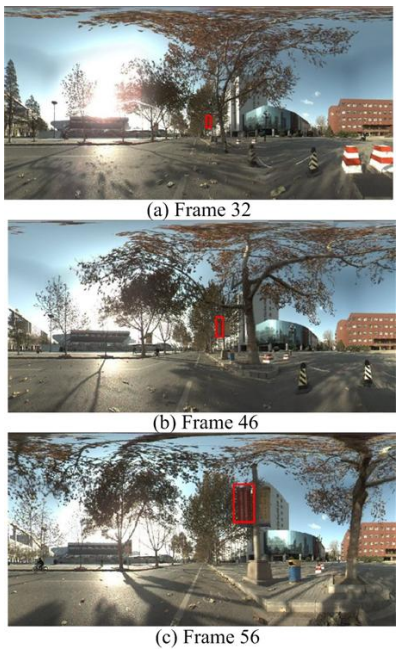

Figure 18. BD1 Sequnece

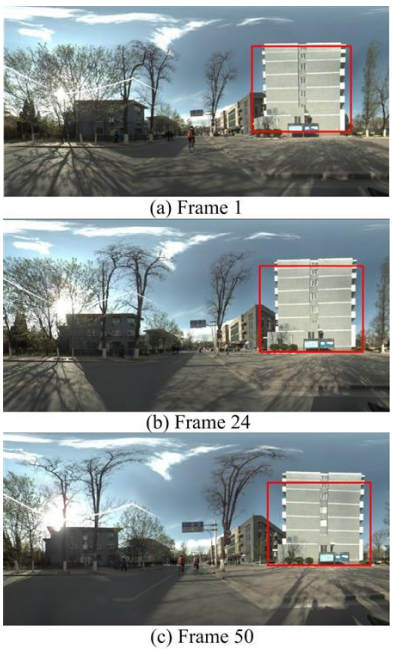

Figure 19. Building Sequence